MULTISTAGE RESOURCE SCHEDULING

CROSS-REFERENCE TO RELATED APPLICATION

[0001] This application claims the benefit of priority of Singapore Patent Application No. 10202400919W filed on 28 March 2024, the content of which being hereby incorporated by reference in its entirety for all purposes.

TECHNICAL FIELD

[0002] The present invention generally relates to a method of multistage resource scheduling, and a system thereof.

BACKGROUND

[0003] In recent years, for example, with the increasingly prevalent integration of renewable energy in energy algorithms around the world, concepts such as "smart microgrids", "integrated energy algorithms" and "distributed energy algorithms" have been proposed and heavily researched to address the challenges posed by the intermittent nature of renewable energies. Many studies have explored the design of energy scheduling methods or algorithms to manage the uncertainties and address the challenges of real-time energy scheduling. Most of these studies focus on multi-timescale design with both day-ahead and real-time scheduling to manage uncertainties and achieve an optimal economic schedule (e.g., resource dispatch or allocation). A two-stage resource scheduling method or algorithm is generally adopted in such works. However, such resource scheduling algorithms would lead to sub-optimality if the forecasts made in the first stage (e.g., day-ahead stage) vary greatly from the ground truth realized in the second stage (e.g., real-time stage). To improve on the resource scheduling algorithm, an additional stage may be inserted between the two stages to adjust the resource allocation solutions or decisions in the generated resource schedule from the first stage, thus forming a three-stage algorithm. However, such a conventional additional stage does not perform resource schedule optimization and its primary function is to track the first stage resource schedule as closely as possible, adjusting only when constraints are updated.

[0004] Accordingly, like the two-stage algorithms, in the three-stage algorithm of most existing works, the resource scheduling solutions or decisions derived from the upper stage are still respected and tracked closely in the lower stages. This follows from the conventional teaching or assumption that the stage with a longer prediction horizon (e.g., the first stage, such

as a day-ahead stage) considers more information and therefore produces better resource scheduling decisions. Thus, it is conventionally understood that the resource scheduling decisions made in the upper stage (which has a longer prediction horizon) need to be closely tracked by the lower stages for optimal operation. The aforementioned assumption may be true when there is access to perfect information. However, the premise of perfect knowledge does not exist in real-world applications. Furthermore, forecasts made further in time may generally associated with greater forecast errors. Therefore, in practice, utilizing information forecasted over the longer prediction horizon of the first or upper stage to generate resource schedule may be riddled with errors. Hence, in practice, the resource schedule generated in the first or upper stage may be sub-optimal and may not be worth being tracked closely in the lower stages.

[0005] Furthermore, resource scheduling algorithms with rolling horizon optimization implemented typically simply utilize a fixed and predefined (e.g., arbitrarily chosen) length of prediction horizon, without justification. In this regard, since the prediction horizon length impacts the amount of information considered to generate a resource schedule, the fixed and predefined prediction horizon length used in the rolling horizon optimization may impact the optimality of the resource schedule generated, resulting in the resource schedule generated being sub-optimal.

[0006] A need therefore exists to provide a method of multistage resource scheduling, and as well as a system thereof, that seeks to overcome, or at least ameliorate, one or more deficiencies in conventional multistage resource scheduling methods, and more particularly, with improved resource scheduling for enhancing the optimality of the resource schedule generated, thereby resulting in improved efficiencies and effectiveness in managing resources. It is against this background that the present invention has been developed.

SUMMARY

[0007] According to a first aspect of the present invention, there is provided a method of multistage resource scheduling, the method comprising:

determining, at a first stage, a first stage resource schedule across a first stage scheduling horizon, comprising a series of timesteps, for a plurality of resource parameters based on a first optimization function; and

determining, at a second stage, a second stage resource schedule across a second stage scheduling horizon, comprising a series of timesteps, for the plurality of resource parameters

based on a second optimization function, comprising, for each timestep, in turn, of the series of timesteps of the second stage scheduling horizon:

determining a resource schedule across a prediction horizon with respect to the timestep, comprising a subseries of timesteps, starting with the timestep, of the series of timesteps of the second stage scheduling horizon, for the plurality of resource parameters based on the second optimization function; and

storing values of the resource schedule determined for the plurality of resource parameters for a beginning timestep of the subseries of timesteps as determined values for the plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon,

wherein for each of one or more timesteps or each of one or more intervals of timesteps of the series of timesteps of the second stage scheduling horizon, respectively, the method further comprises determining and setting a length of the prediction horizon for the above-mentioned determining the resource schedule across the prediction horizon.

[0008] According to a second aspect of the present invention, there is provided a system for multistage resource scheduling, the system comprising:

at least one memory; and

at least one processor communicatively coupled to the at least one memory and configured to:

determine, at a first stage, a first stage resource schedule across a first stage scheduling horizon, comprising a series of timesteps, for a plurality of resource parameters based on a first optimization function; and

determine, at a second stage, a second stage resource schedule across a second stage scheduling horizon, comprising a series of timesteps, for the plurality of resource parameters based on a second optimization function, comprising, for each timestep, in turn, of the series of timesteps of the second stage scheduling horizon:

determining a resource schedule across a prediction horizon with respect to the timestep, comprising a subseries of timesteps, starting with the timestep, of the series of timesteps of the second stage scheduling horizon, for the plurality of resource parameters based on the second optimization function; and

storing values of the resource schedule determined for the plurality of resource parameters for a beginning timestep of the subseries of timesteps as determined values for the

plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon,

wherein for each of one or more timesteps or each of one or more intervals of timesteps of the series of timesteps of the second stage scheduling horizon, respectively, the at least one processor is further configured to determine and set a length of the prediction horizon for the above-mentioned determining the resource schedule across the prediction horizon.

[0009] According to a third aspect of the present invention, there is provided a computer program product, embodied in one or more non-transitory computer-readable storage mediums, comprising instructions executable by at least one processor to perform the method of multistage resource scheduling according to the above-mentioned first aspect of the present invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] Embodiments of the present invention will be better understood and readily apparent to one of ordinary skill in the art from the following written description, by way of example only, and in conjunction with the drawings, in which:

FIG. 1 depicts a schematic flow diagram of a method of multistage resource scheduling, according to various embodiments of the present invention;

FIG. 2 depicts a schematic block diagram of a system for multistage resource scheduling, according to various embodiments of the present invention;

FIG. 3 depicts a schematic block diagram of an example three-stage resource scheduling method having a day-ahead (DA) stage, an intraday (ID) stage and a real-time (RT) stage, according to various embodiments of the present invention;

FIG. 4 illustrates the rolling horizon optimization implemented at the ID stage with the adaptive prediction horizon (APH) and the interstage re-optimization procedures or processes of the resource scheduling method according to various example embodiments of the present invention;

FIG. 5 depicts a flow diagram of the APH procedure according to various example embodiment of the present invention when implemented on an example real-time resource scheduling problem;

FIG. 6 depicts a flow diagram of the interstage re-optimization procedure according to various example embodiments of the present invention;

FIG. 7A depicts Table 2 presenting values of fixed parameters and constraints used in simulation;

FIG. 7B depicts Table 3 presenting key summary of three resource scheduling methods being evaluated;

FIG. 7C depicts Table 4 presenting respective resource scheduling methods' power allocation deviation from optimal baseline;

FIG. 7D depicts Table 5 presenting cost index of respective resource scheduling methods relative to optimal baseline;

FIG. 7E depicts Table 6 presenting resource cost index of respective resource scheduling methods relative to optimal baseline; and

FIG. 7F depicts Table 7 presenting carbon emission cost index of respective algorithms relative to optimal baseline.

DETAILED DESCRIPTION

[0011] Various embodiments of the present invention provide a method of multistage resource scheduling, and a system thereof.

[0012] As described in the background, in conventional multistage resource scheduling methods or algorithms, based on the conventional teaching or assumption that the stage with a longer prediction horizon (e.g., the first stage, such as a day-ahead stage) considers more information and therefore produces better resource scheduling decisions, the resource scheduling solutions or decisions derived from the upper stage are still respected and tracked closely in the lower stages. Furthermore, conventional resource scheduling algorithms with rolling horizon optimization implemented typically simply utilize a fixed and predefined (e.g., arbitrarily chosen) length of prediction horizon, without justification. As a result, various embodiments of the present invention note that the optimality of the resource schedules generated by such conventional multistage resource scheduling methods or algorithms are undesirably impacted, resulting in the resource schedules generated being sub-optimal, which in turn results in inefficiencies and ineffectiveness in managing resources. In particular, various embodiments of the present invention note that in such conventional multistage resource scheduling methods, there is no re-optimization performed in any lower stage for evaluating or modifying the resource scheduling decisions of the first or upper stage. Furthermore, various embodiments of the present invention note that the particular length of the prediction horizon impacts both the accuracy of forecasts generated and the amount of information considered to

generate a resource schedule. In this regard, various embodiments of the present invention provide a method of multistage resource scheduling, and as well as a system thereof, that seeks to overcome, or at least ameliorate, one or more deficiencies in conventional multistage resource scheduling methods, and more particularly, with improved resource scheduling for enhancing the optimality of the resource schedule generated, thereby resulting in improved efficiencies and effectiveness in managing resources.

[0013] FIG. 1 depicts a schematic flow diagram of a method 100 of multistage resource scheduling, according to various embodiments of the present invention. The method 100 comprises: determining (106), at a first stage, a first stage resource schedule across a first stage scheduling horizon, comprising a series of timesteps, for a plurality of resource parameters based on a first optimization function; and determining (108), at a second stage, a second stage resource schedule across a second stage scheduling horizon, comprising a series of timesteps, for the plurality of resource parameters based on a second optimization function, comprising, for each timestep, in turn, of the series of timesteps of the second stage scheduling horizon: determining a resource schedule across a prediction horizon with respect to the timestep, comprising a subseries of timesteps, starting with the timestep, of the series of timesteps of the second stage scheduling horizon, for the plurality of resource parameters based on the second optimization function; and storing values of the resource schedule determined for the plurality of resource parameters for a beginning timestep (i.e., the very first timestep) of the subseries of timesteps as determined values for the plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon. In particular, for each of one or more timesteps or each of one or more intervals of timesteps of the series of timesteps of the second stage scheduling horizon, respectively, the method 100 further comprises determining and setting (at 110) a length of the prediction horizon for the above-mentioned determining the resource schedule across the prediction horizon.

[0014] Accordingly, the method 100 of multistage resource scheduling according to various embodiments of the present invention is able to improve resource scheduling for enhancing the optimality of the resource schedule generated. In particular, according to the method 100, when determining the second stage resource schedule across the second stage scheduling horizon, the length of the prediction horizon used in the rolling horizon optimization is not predefined or fixed across the second stage scheduling horizon but is advantageously adaptive or dynamic. In particular, for each of a number of timesteps or intervals of timesteps of the second stage scheduling horizon, the length of the prediction horizon is specifically determined and set for

the timestep or the interval of timesteps. Therefore, according to the method 100, the length of the prediction horizon used in the rolling horizon optimization is advantageously adjusted or optimized over the second stage scheduling horizon, which has been found to enhance the optimality of the resource schedule generated by the second stage. These advantages or technical effects, and/or other advantages or technical effects, will become more apparent to a person skilled in the art as the method 100 of multistage resource scheduling, as well as the corresponding system for multistage resource scheduling, is described in more detail according to various embodiments and example embodiments of the present invention.

[0015] In various embodiments, for the above-mentioned each timestep, in turn, of the series of timesteps of the second stage scheduling horizon: the above-mentioned determining the resource schedule across the prediction horizon with respect to the timestep comprises generating forecasted values for the subseries of timesteps of the prediction horizon for the plurality of resource parameters, and the method 100 further comprises storing forecasted values for the plurality of resource parameters for the beginning timestep (i.e., the first timestep) of the subseries of timesteps as forecasted values for the plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon. Furthermore, for the above-mentioned each of the one or more timesteps or the above-mentioned each of the one or more intervals of timesteps of the second stage scheduling horizon, respectively, the length of the prediction horizon for the above-mentioned determining the resource schedule across the prediction horizon is determined based on an error measure between past forecasted values stored and measured values for the plurality of resource parameters for one or more past timesteps of the second stage scheduling horizon.

[0016] In various embodiments, the length of the prediction horizon for the above-mentioned determining the resource schedule across the prediction horizon is determined using a prediction horizon length machine learning model based on the error measure between the past forecasted values stored and the measured values for the plurality of resource parameters for the one or more past timesteps of the second stage scheduling horizon.

[0017] In various embodiments, for the above-mentioned each timestep, in turn, of the series of timesteps of the second stage scheduling horizon, the second optimization function comprises a schedule deviation optimization function and a resource cost optimization function. The second optimization function is configured to minimize, over the prediction horizon with respect to the timestep, a deviation between the resource schedule determined across the prediction horizon with respect to the timestep and a corresponding portion of the first stage

resource schedule. The resource cost optimization function is configured to minimize, over the prediction horizon with respect to the timestep, a cost objective associated with the plurality of resource parameters.

[0018] In various embodiments, the above-mentioned determining (at 106), at the first stage, the first stage resource schedule across the first stage scheduling horizon comprises generating and storing forecasted values for the series of timesteps of the first stage scheduling horizon for the plurality of resource parameters. As described hereinbefore, for the above-mentioned each timestep, in turn, of the series of timesteps of the second stage scheduling horizon: the above-mentioned determining the resource schedule across the prediction horizon with respect to the timestep comprises generating forecasted values for the subseries of timesteps of the prediction horizon for the plurality of resource parameters, and the method 100 comprises storing forecasted values for the plurality of resource parameters for the beginning timestep of the subseries of timesteps as forecasted values for the plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon. In this regard, for the above-mentioned each of one or more timesteps or the above-mentioned each of one or more intervals of timesteps of the second stage scheduling horizon, the method 100 further comprises: determining and setting one or more values of one or more optimization weight parameters, respectively, based on past forecasted values stored for the plurality of resource parameters for the first stage and past forecasted values stored for the plurality of resource parameters for the second stage. The second optimization function is weighted by the one or more optimization weight parameters. By weighting the second optimization function (comprising the schedule deviation optimization function and the resource cost optimization function) based on the one or more optimization weight parameters and determining and setting one or more values of the one or more optimization weight parameters, respectively, based on past forecasted values stored for the plurality of resource parameters for the first stage and past forecasted values stored for the plurality of resource parameters for the second stage, the method 100 is able to further improve resource scheduling for further enhancing the optimality of the resource schedule generated. In particular, the second stage resource schedule across the second stage scheduling horizon is not simply determined to closely track the first stage resource schedule, but the degree of tracking the first resource schedule (or between the first stage cost objective and the second stage cost objective) is advantageously dynamic, adaptive or optimized based on the one or more optimization weight parameters, which has been found to further enhance the optimality of the resource schedule generated by the second stage.

[0019] In various embodiments, the one or more values of the one or more optimization weight parameters are determined based on a first error measure and a second error measure. The first error measure is between the past forecasted values stored and measured values for the plurality of resource parameters for one or more past timesteps of the first stage scheduling horizon. The second error measure is between the past forecasted values stored and measured values for the plurality of resource parameters for the one or more past timesteps of the second stage scheduling horizon.

[0020] In various embodiments, the second optimization function is weighted by the one or more optimization weight parameters such that a larger first error measure relative to the second error measure results in a larger weighting of the resource cost optimization function relative to the schedule deviation optimization function, and a smaller first error measure relative to the second error measure results in a smaller weighting of the resource cost optimization function relative to the schedule deviation optimization function.

[0021] In various embodiments, the method 100 further comprises: determining, at a third stage, a third stage resource schedule across a third stage scheduling horizon, comprising a series of timesteps, for the plurality of resource parameters based on a third optimization function. The third optimization function is configured to minimize a deviation between the third stage resource schedule and the second stage resource schedule subject to one or more constraints.

[0022] In various embodiments, the first stage, the second stage and the third stage correspond to a day-ahead stage, an intraday stage and a real-time stage, respectively.

[0023] In various embodiments, the above-mentioned determining and setting (at 110) the length of the prediction horizon and the above-mentioned determining and setting the one or more values of the one or more optimization weight parameters are performed at a beginning of each interval of a plurality of consecutive intervals of timesteps of the series of timesteps of the second stage scheduling horizon.

[0024] FIG. 2 depicts a schematic block diagram of a system 200 for multistage resource scheduling, according to various embodiments of the present invention, corresponding to the above-mentioned method 100 of multistage resource scheduling as described hereinbefore with reference to FIG. 1 according to various embodiments of the present invention. The system 200 comprises: at least one memory 202; and at least one processor 204 communicatively coupled to the at least one memory 202 and configured to perform the method 100 of multistage resource scheduling according to various embodiments of the present invention. Accordingly, the at least

one processor 204 is configured to: determine, at a first stage, a first stage resource schedule across a first stage scheduling horizon, comprising a series of timesteps, for a plurality of resource parameters based on a first optimization function; and determine, at a second stage, a second stage resource schedule across a second stage scheduling horizon, comprising a series of timesteps, for the plurality of resource parameters based on a second optimization function, comprising, for each timestep, in turn, of the series of timesteps of the second stage scheduling horizon: determining a resource schedule across a prediction horizon with respect to the timestep, comprising a subseries of timesteps, starting with the timestep, of the series of timesteps of the second stage scheduling horizon, for the plurality of resource parameters based on the second optimization function; and storing values of the resource schedule determined for the plurality of resource parameters for a beginning timestep of the subseries of timesteps as determined values for the plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon. In particular, for each of one or more timesteps or each of one or more intervals of timesteps of the series of timesteps of the second stage scheduling horizon, respectively, the at least one processor 204 is further configured to determine and set a length of the prediction horizon for the above-mentioned determining the resource schedule across the prediction horizon.

[0025] It will be appreciated by a person skilled in the art that the at least one processor 204 may be configured to perform various functions or operations through set(s) of instructions (e.g., software modules) executable by the at least one processor 204 to perform various functions or operations. Accordingly, as shown in FIG. 2, the system 200 may comprise: a first stage resource schedule module (or a first stage resource schedule circuit) 206 configured to determine, at a first stage, a first stage resource schedule across a first stage scheduling horizon, comprising a series of timesteps, for a plurality of resource parameters based on a first optimization function; and a second stage resource schedule module (or a second stage resource schedule circuit) 208 configured to determine, at a second stage, a second stage resource schedule across a second stage scheduling horizon, comprising a series of timesteps, for the plurality of resource parameters based on a second optimization function, comprising, for each timestep, in turn, of the series of timesteps of the second stage scheduling horizon: determining a resource schedule across a prediction horizon with respect to the timestep, comprising a subseries of timesteps, starting with the timestep, of the series of timesteps of the second stage scheduling horizon, for the plurality of resource parameters based on the second optimization function; and storing values of the resource schedule determined for the plurality of resource

parameters for a beginning timestep of the subseries of timesteps as determined values for the plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon; and a prediction horizon length determining and setting module (or a prediction horizon length determining and setting circuit) 210 configured to, for each of one or more timesteps or each of one or more intervals of timesteps of the series of timesteps of the second stage scheduling horizon, respectively, determine and set a length of the prediction horizon for the above-mentioned determining the resource schedule across the prediction horizon.

[0026] It will be appreciated by a person skilled in the art that the above-mentioned modules are not necessarily separate modules, and two or more modules may be realized by or implemented as one functional module (e.g., a circuit or a software program) as desired or as appropriate without deviating from the scope of the present invention. For example, two or more of the first stage resource schedule module 206, the second stage resource schedule module 208 and the prediction horizon length determining and setting module 210 may be realized (e.g., compiled together) as one executable software program (e.g., software application), which for example may be stored in the at least one memory 202 and executable by the at least one processor 204 to perform the corresponding functions or operations as described herein according to various embodiments of the present invention.

[0027] In various embodiments, the system 200 for multistage resource scheduling corresponds to the method 100 of multistage resource scheduling as described hereinbefore with reference to FIG. 1, therefore, various operations, functions or steps configured to be performed by the least one processor 204 may correspond to various operations, functions or steps of the method 100 described hereinbefore according to various embodiments, and thus need not be repeated with respect to the system 200 for clarity and conciseness. In other words, various embodiments described herein in context of methods (e.g., the method 100 of multistage resource scheduling) are analogously valid for the corresponding systems or devices (e.g., the system 200 for multistage resource scheduling), and vice versa. For example, in various embodiments, the at least one memory 202 may have stored therein the first stage resource schedule module 206, the second stage resource schedule module 208 and/or the prediction horizon length determining and setting module 210, which respectively correspond to various operations, functions or steps of the method 100 of multistage resource scheduling as described hereinbefore according to various embodiments, which are executable by the at least one processor 204 to perform the corresponding operations, functions or steps as described herein.

[0028] A computing system, a controller, a microcontroller or any other system providing a processing capability may be provided according to various embodiments in the present invention. Such a system may be taken to include one or more processors and one or more computer-readable storage mediums. For example, the system 200 for multistage resource scheduling described hereinbefore may include at least one processor 204 and at least one computer-readable storage medium (or memory) 202 which are for example used in various processing carried out therein as described herein. A memory or computer-readable storage medium used in various embodiments may be a volatile memory, for example a DRAM (Dynamic Random Access Memory) or a non-volatile memory, for example a PROM (Programmable Read Only Memory), an EPROM (Erasable PROM), EEPROM (Electrically Erasable PROM), or a flash memory, e.g., a floating gate memory, a charge trapping memory, an MRAM (Magnetoresistive Random Access Memory) or a PCRAM (Phase Change Random Access Memory).

[0029] In various embodiments, a "circuit" may be understood as any kind of a logic implementing entity, which may be special purpose circuitry or a processor executing software stored in a memory, firmware, or any combination thereof. Thus, in an embodiment, a "circuit" may be a hard-wired logic circuit or a programmable logic circuit such as a programmable processor, e.g., a microprocessor (e.g., a Complex Instruction Set Computer (CISC) processor or a Reduced Instruction Set Computer (RISC) processor). A "circuit" may also be a processor executing software, e.g., any kind of computer program, e.g., a computer program using a virtual machine code, e.g., Java. Any other kind of implementation of various functions or operations may also be understood as a "circuit" in accordance with various other embodiments. Similarly, a "module" may be a portion of a system according to various embodiments in the present invention and may encompass a "circuit" as above, or may be understood to be any kind of a logic-implementing entity therefrom.

[0030] Some portions of the present disclosure may be explicitly or implicitly presented in terms of algorithms and functional or symbolic representations of operations on data within a computer memory. These algorithmic descriptions and functional or symbolic representations are the means used by those skilled in the data processing arts to convey most effectively the substance of their work to others skilled in the art. An algorithm may be, and generally, conceived to be a self-consistent sequence of steps leading to a desired result.

[0031] The present specification also discloses a system (e.g., which may also be embodied as one or more devices or apparatuses), such as the system 200 for multistage resource

scheduling, for performing various operations, functions or steps of various methods described herein. Such a system may be specially constructed for the required purposes or may comprise a general purpose computer system selectively activated or reconfigured by a computer program stored in the computer system. In general, various algorithms that may be presented herein are not limited to being implemented or executed by any particular computer system. Alternatively, the construction of more specialized computer system to perform various operations, functions or steps of various methods described herein may be provided as desired or as appropriate without going beyond the scope of the present invention.

[0032] In addition, the present specification also at least implicitly discloses computer program(s) or software/functional module(s), in that it would be apparent to a person skilled in the art that various operations, functions or steps of various methods described herein may be put into effect by computer code. The computer program(s) is not intended to be limited to any particular programming language and implementation thereof, and it will be appreciated by a person skilled in the art that a variety of programming languages and coding thereof may be used to implement the computer program(s). Moreover, the computer program(s) is not intended to be limited to any particular control flow as there are a variety of programming languages which can use different control flows. It will be appreciated by a person skilled in the art that a computer program may be stored on any computer-readable storage medium (non-transitory computer-readable storage medium), such as but not limited to, a magnetic disk, an optical disk or a memory chip. For example, a computer program stored on a computer-readable storage medium may be loaded and executed on a computer system to implement various operations, functions or steps of various methods described herein according to various embodiments of the present invention.

[0033] Accordingly, in various embodiments, there is provided a computer program product, embodied in one or more computer-readable storage mediums (non-transitory computer-readable storage medium), comprising instructions (e.g., the first stage resource schedule module 206, the second stage resource schedule module 208 and/or the prediction horizon length determining and setting module 210) executable by one or more computer processors to perform a method 100 of multistage resource scheduling as described hereinbefore with reference to FIG. 1 according to various embodiments of the present invention. Accordingly, various computer programs or software modules described herein may be stored in a computer program product receivable by a system therein, such as the system 200 as shown in FIG. 2, for execution by at least one processor 204 of the system 200 to perform

various operations, functions or steps of various methods described herein according to various embodiments of the present invention.

[0034] It will be appreciated by a person skilled in the art that various modules described herein (e.g., the first stage resource schedule module 206, the second stage resource schedule module 208 and/or the prediction horizon length determining and setting module 210) may be software module(s) realized by computer program(s) or set(s) of instructions executable by a computer processor to perform various functions or operations. Various modules described herein (e.g., the first stage resource schedule module 206, the second stage resource schedule module 208 and/or the prediction horizon length determining and setting module 210), together with the at least one processor 204 and the at least one memory 202, may also be implemented as hardware module(s) being functional hardware unit(s) designed to perform various functions or operations. More particularly, in the hardware sense, a module is a functional hardware unit designed for use with other components or modules. For example, a module may be implemented using discrete electronic components, or it may form a portion of an entire electronic circuit such as an Application Specific Integrated Circuit (ASIC) or a Field Programmable Gate Array (FPGA). Numerous other possibilities exist. It will also be appreciated by a person skilled in the art that a combination of hardware and software modules may be implemented. Furthermore, various operations, functions or steps of various methods described herein may be performed in parallel rather than sequentially as desired or as appropriate (e.g., as long as it does not render the method(s) inoperable or unsatisfactory for its intended purpose).

[0035] It will be appreciated by a person skilled in the art that the terminology used herein is for the purpose of describing various embodiments only and is not intended to be limiting of the present invention. As used herein, the singular forms "a", "an" and "the" are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will be further understood that the terms "comprises" and/or "comprising," when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof.

[0036] Any reference to an element or a feature herein using a designation such as "first", "second" and so forth does not limit the quantity or order of such elements or features, unless stated or the context requires otherwise. For example, such designations may be used herein as a convenient way of distinguishing between two or more elements or instances of an element.

Thus, a reference to first and second elements does not necessarily mean that only two elements can be employed, or that the first element must precede the second element, unless stated or the context requires otherwise. In addition, a phrase referring to "at least one of" a list of items refers to any single item therein or any combination of two or more items therein.

[0037] In order that the present invention may be readily understood and put into practical effect, various example embodiments of the present invention will be described hereinafter by way of examples only and not limitations. It will be appreciated by a person skilled in the art that the present invention may, however, be embodied in various different forms or configurations and should not be construed as limited to the example embodiments set forth hereinafter. Rather, these example embodiments are provided so that this disclosure will be thorough and complete, and will fully convey the scope of the present invention to those skilled in the art.

[0038] In particular, for better understanding of the present invention and without limitation or loss of generality, various example embodiments of the present invention will now be described with respect to a method of multistage resource scheduling being a three-stage resource scheduling method and more specifically, a first stage being a day-ahead stage, a second stage being an intraday stage and a third stage being a real-time stage. It will be understood by a person skilled in the art that the method of multistage resource scheduling is not limited to such a specific three-stage resource scheduling method and various types of multistage resource scheduling may be implemented in a wide range of practical applications without going beyond the scope of the present invention as long as it involves a first stage and one or more subsequent stages of resource scheduling, such as but not limited to: (1) inventory planning - to optimize inventory levels and facilitate a smooth supply chain, where the first, second and third stages may be year-ahead planning, quarterly planning and monthly planning, respectively; (2) autonomous driving - to optimize the operation of autonomous vehicles, where the first, second and third stages may be long-term overall route planning, short-term route adjustment, and real-time navigation, respectively; (3) various resource management tasks, for example, resource allocation such as staff, equipment, and budget in healthcare, manufacturing, etc., where the first, second and third stages may be annual planning, quarterly planning and daily planning, respectively; and (4) multi-year investment strategy of various technologies for policymaking, to plan and allocate resources for the development and implementation of various technologies over different time horizons, where the first, second and third stages may be multi-year planning, yearly planning and monthly planning, respectively.

[0039] Various example embodiments provide a method (e.g., an algorithm) for multistage real-time resource scheduling. As an illustrative example, the multistage resource scheduling method comprises day-ahead (DA), intraday (ID), and real-time (RT) stages, that seeks to optimally manage or allocate multiple resources (e.g., energy and material resources) to meet various demands, thereby minimizing resource costs and environmental impacts (e.g., carbon emission costs). The method for multistage resource scheduling may be applied for managing or allocating resources in a wide range of practical applications, such as multi-energy (e.g., grid, solar, wind, hydro, heat, and so on) and multi-resource (e.g., water, emissions, materials, wastes, and so on) and jobs/tasks and/or machine scheduling, manpower allocation and planning, and so on. The method for multistage resource scheduling may also be applied to process control that is based on model predictive control / receding horizon optimization / rolling horizon optimization (e.g., heating, ventilation, and air-conditioning (HVAC) control, autonomous driving, plant control, and so on).

[0040] FIG. 3 depicts a schematic block diagram of a three-stage resource scheduling method (or algorithm) 300 having a first stage being a day-ahead (DA) stage, a second stage being an intraday (ID) stage and a third stage being a real-time (RT) stage, for addressing one or more deficiencies in conventional multistage resource scheduling methods, such as those described hereinbefore in the background, and more particular, with improved resource scheduling such as improved management of renewable energy source penetrated grids.

[0041] The DA stage, ID stage, and RT stage has timestep t_{DA}, t_{ID}, and t, respectively. For example, the scale of the timesteps' duration may be set by the resource scheduling algorithm's designer. Generally, in existing literature, the scale of each stage's timestep's duration in a multistage scheduling algorithm may become finer down the stages. For example, the duration of timestep t_{DA}, t_{ID} may both be in terms of hours and timestep t may be in terms of minutes. For example, both the DA and ID stages may have the same timestep duration, that is, $\Delta t_{DA} = \Delta t_{ID}$.

[0042] According to various example embodiments, in the three-stage resource scheduling method 300, there are two key modules or procedures implemented at the ID stage, which may herein be referred to as adaptive prediction horizon (APH) and interstage re-optimization. An example process flow of the resource scheduling method 300, including the APH and interstage re-optimization, is shown in FIG. 3, whereby the APH and interstage re-optimization modules, along with the intraday (ID) stage, are enclosed by a dashed box. In various example embodiments, the APH module or process is configured to decide on the optimal prediction

horizon length to use, which is herein denoted as PH. In various example embodiments, the interstage re-optimization module or process involves one or more optimization weight parameters, such as weight parameters γ and δ. The APH and interstage re-optimization processes will be described in detail later below according to various example embodiments of the present invention.

[0043] For the sake of clarity in understanding, example relationships between the timesteps of different scheduling stages of the resource scheduling method 300 will now be described according to various example embodiments of the present invention. For example, it is assumed that the duration of the day-ahead stage's timestep Δt_{DA} and the duration of the intraday stage's timestep Δt_{ID} share the same duration of 1 hour, and the duration of the RT stage's timestep Δt is 15 minutes. The resource scheduling begins at the first RT timestep $t = 0$, which corresponds to the first timesteps t_{DA}, t_{ID} of both DA and ID stages, that is, $t_{DA} = t_{ID} = 0$. Therefore, four RT timesteps correspond to one ID timestep ($\Delta t_{ID} = 4\Delta t$), $t = 4$ corresponds to $t_{ID} = 1$, and since, $\Delta t_{DA} = \Delta t_{ID}$, $t_{DA} = 1$ corresponds to $t_{ID} = 1$. Furthermore, as can be seen in FIG. 3 according to various example embodiments, both APH and interstage re-optimization procedures in the ID stage may be initiated or activated after every stipulated or predefined interval of timesteps, to update the associated parameters (e.g., parameters PH, γ and δ). Accordingly, for each interval (e.g., interval of K =16) of timesteps of the series of timesteps of the ID scheduling horizon, respectively, the method 300 further comprises determining and setting a length of the prediction horizon for determining the resource schedule across the prediction horizon. For example, if the APH and interstage re-optimization procedures are stipulated to activate at the start of every four t_{ID}, this corresponds to an RT timestep interval of K =16. This means that the APH and interstage re-optimization procedures are activated when $t \% K = 0$, which is equivalent to $t \% 16 = 0$.

[0044] The resource scheduling method 300 will now be described in further detail with reference to FIG. 3 according to various example embodiments of the present invention. The resource scheduling method 300 comprises: determining, at the DA stage, a DA resource schedule across a DA scheduling horizon, comprising a series of timesteps, for a plurality of resource parameters based on a first optimization function; and determining, at the ID stage, an ID resource schedule across an ID scheduling horizon (e.g., the same duration as the DA scheduling horizon), comprising a series of timesteps, for the plurality of resource parameters based on a second optimization function, comprising, for each timestep, in turn, of the series of timesteps of the ID scheduling horizon: determining a resource schedule across a prediction

horizon with respect to the timestep (i.e., based on rolling horizon optimization), comprising a subseries of timesteps, starting with the timestep, of the series of timesteps of the ID scheduling horizon, for the plurality of resource parameters based on the second optimization function; and storing values of the resource schedule determined for the plurality of resource parameters for a beginning timestep (i.e., the very first timestep) of the subseries of timesteps as determined values for the plurality of resource parameters for the timestep of the series of timesteps of the ID scheduling horizon.

[0045] In various example embodiments, the resource scheduling first starts with the DA stage. For example, on the day before the scheduling date (the scheduling date corresponding to $t = t_{ID} = 0$), future or forecasted values of the respective stochastic parameters (i.e., the resource parameters) are forecasted across the scheduling date's scheduling horizon ($t_{DA} = 0$ to $t_{DA} = T$) (corresponding to the DA predicted data shown in FIG. 3) and used to generate a DA resource schedule across the scheduling horizon. In this regard, the DA resource schedule may be generated based on the forecasted values of the stochastic parameters using rolling horizon optimization across the scheduling horizon. For example, as shown in FIG. 3, the rolling horizon optimization may be performed based on an optimization function (e.g., corresponding to the first optimization function as described hereinbefore) such as by solving a nonlinear programming (NLP) optimization problem in the DA stage across the scheduling horizon. In this regard, as will be described later below, the generated DA resource schedule is referenced and may be improved/adjusted on by the subsequent stages closer to the real-time operations.

[0046] On the scheduling date, the resource scheduling begins at $t = t_{ID} = 0$. In various example embodiments, the initial values of the parameters utilized in the method 300 may be set in the following manner. The length of the beginning prediction horizon (i.e., the first prediction horizon in the ID scheduling horizon, which may be denoted as $PH_{initial}$) used may be chosen (e.g., arbitrarily chosen) from a specified range or set of prediction horizon lengths (this set of prediction horizon lengths is used to train the APH procedure's machine learning module (e.g., a classifier) as will be described later below according to various example embodiments of the present invention). For example, the initial values of weight parameters γ and δ are initialized with equal values (e.g., $\gamma = \delta = 0.5$) as no assumptions are made on the relative performance (e.g., error performance) between the resource schedules generated by the DA and ID stages, respectively.

[0047] At each timestep t_{ID}, the past measured values of the stochastic parameters are used to generate new forecasts of the stochastic parameters. For example, for the very first timestep t_{ID} whereby there may be no past measured values of the stochastic parameters available, the day-ahead forecasts generated by the day-ahead (DA) stage may be used. An optimization (rolling horizon optimization) that uses both the DA resource schedule and the newly generated stochastic parameters' forecasts may then be carried out over the prediction horizon having a length determined and set by the APH module. The optimization is performed based on an optimization function (e.g., corresponding to the second optimization function as described hereinbefore) such as by solving a nonlinear programming (NLP) optimization problem in the ID stage based on parameters PH, γ and δ. For example, the optimization problem is the minimization of the combined weighted sum of (1) the minimization of a schedule deviation objective (between the ID cost objective and the DA cost objective) (weighted by weight parameter γ) (e.g., corresponding to the schedule deviation optimization function as described hereinbefore) and (2) the minimization of the ID cost objective (weighted by weight parameter δ) (e.g., corresponding to the resource cost optimization function as described hereinbefore). For example, the ID cost objective may refer to the sum of resource cost and carbon emission cost. For each prediction horizon, only the determined values of the resource schedule for the first timestep and the forecasted values for the first timestep are retained or stored and the rest is discarded.

[0048] At the start of every stipulated timestep interval K, both the APH and interstage procedures are initiated or activated to update the parameter PH used in the ID stage and the weight parameters γ and δ, which controls whether to closely track the resource schedule generated in the DA stage or adjust them with the ID cost objective associated with the ID stage so as to obtain re-optimized solutions (ID resource schedule) in the ID stage. The final retained resource scheduling solutions (ID resource schedule across the ID scheduling horizon) thus constitutes the adjusted or optimized resource schedule generated in the ID stage.

[0049] Finally, at the RT stage, the determined values (e.g., corresponding to resource dispatch decisions or allocations) of the adjusted resource schedule from the ID stage are closely tracked. For example, the optimization problem of the RT stage may be to minimize the deviation of determined values from the resource schedule generated in the ID stage (e.g., corresponding to the third optimization function described hereinbefore). The measured values of the stochastic parameters in the RT stage may be used to update constraints of the optimization problem. The solutions of the optimization problem at the RT stage may then form

the final resource schedule for the resources (e.g., corresponding to the final scheduling dispatch decisions) over the scheduling horizon.

[0050] As an illustrative example, FIG. 4 illustrates the rolling horizon optimization implemented at the ID stage with the APH and re-optimization processes of the method 300 according to various example embodiments of the present invention.

[0051] As shown in FIG. 4, at the beginning of the ID stage, an optimization is carried out across the initialized prediction horizon, $PH_{initial}$. As can be seen from FIG. 4, at each optimization performed at the ID stage, only the first timestep's solutions (determined values for the resource parameters for the first timestep) are retained/stored and the remaining solutions for the remaining timesteps are discarded. The resource scheduling then moves on to the next timestep. As can be seen from FIG. 4, when the resource scheduling moves on to the next timestep, the past timesteps' measured and forecasted values (or profiles) of the respective stochastic parameters (resource parameters) are retained/stored (e.g., from $t = 0$ onwards). The retained information are utilized for both APH and interstage re-optimization procedures according to various example embodiments of the present invention.

[0052] At the start of each stipulated timestep interval K, the APH and interstage re-optimization are triggered, as will be described in further detail later below with reference to FIG. 6. In various example embodiments, all the retained past timesteps' measured and forecasted values (or profiles) (t = 0 to t = *current K interval*) of the stochastic parameters (resource parameters) the ID scheduling horizon are utilized in both APH and interstage re-optimization procedures. Accordingly, as shown in FIG. 6, at every stipulated timestep interval K, the prediction horizon length (parameter PH) to be used in the rolling horizon optimization for each timestep of the timestep interval is updated. FIG. 4 also shows that the past timesteps' measured and forecasted profiles of the stochastic parameters (t = 0 to t = *current K interval*) are being utilized for the APH and interstage re-optimization procedures. The above-described rolling horizon optimization is repeated at each timestep until the end of the scheduling horizon, where t = T. For each timestep t_{ID} whereby its prediction horizon exceeds the scheduling horizon during optimization, for example, the optimization may be performed based on ID forecasted values of the stochastic parameters (resource parameters) for timesteps within both the prediction horizon and the ID scheduling horizon and based on DA forecasted values of the stochastic parameters (resource parameters) for corresponding timesteps. That is, the optimization does not require DA or ID forecasted values of the stochastic parameters beyond the scheduling horizon. Accordingly, at the ID stage, the optimization performed at the very

first timestep and at a number of end timesteps of the ID scheduling horizon (the end timesteps whereby at least a portion of its prediction horizon exceed the scheduling horizon) may involve certain or subtle differences than that performed at other timesteps of the ID scheduling horizon. Therefore, in various example embodiments, the above-mentioned series of timesteps of the ID scheduling horizon of the resource scheduling method 300 (and also similarly, the above-mentioned series of timesteps of the second stage scheduling horizon of the resource scheduling method 100) may refer or apply to all timesteps of the ID scheduling horizon (or the second stage scheduling horizon) except the above-mentioned first timestep and the above-mentioned end timesteps.

[0053] The APH will now be described in further detail according to various example embodiments of the present invention.

[0054] In various example embodiments, the APH procedure may utilize a machine learning model, such as a classification model, trained to determine an optimal prediction horizon length. In this regard, at the start of every stipulated timestep interval during resource scheduling (e.g., except for the very first timestep interval as explained above whereby the prediction horizon length may be arbitrarily chosen since there is no available past measured values for the stochastic parameters for optimizing the prediction horizon length), the optimal prediction horizon length may be determined using the classification model. In various example embodiments, features utilized for the APH procedure's classification model may include: the past timesteps' measured values of each stochastic parameter; the past timesteps' forecasted values of each stochastic parameter and each stochastic parameter's past timesteps forecast's root mean squared error (RMSE) (e.g., corresponding to the error measure between past forecasted values stored and measured values for the plurality of resource parameters for one or more past timesteps of the second stage scheduling horizon described hereinbefore), relative to their respective past timesteps' measured values. The label of the classification model is the optimal prediction horizon length to use (e.g., chosen from a specified set of possible/candidate prediction horizon lengths). In various example embodiments, the determination of the optimal prediction horizon length is a multiclass classification problem.

[0055] An example method of training of an APH classification model will now be described according to various example embodiments of the present invention.

[0056] Firstly, a classification algorithm is selected to train on the training examples. In this regard, it will be appreciated by a person skilled in the art that, in general, any suitable machine learning model (e.g., classification algorithm) may be applied as desired or as appropriate, such

as based on the data and context of the problem, and the present invention is not limited to any particular machine learning model. As an illustrative example, a (voting) ensemble classifier comprising (1) balanced random forest classifier, (2) Nu-support vector classification, and (3) linear discriminant analysis may be used. For example, the k-fold cross-validation procedure may be employed to train the ensemble classifier. For example, the ensemble classifier's performance on the training set was used to tune it. The final evaluation is then done on the held out set to evaluate the acceptably tuned classification model.

[0057] To train the classifier model, a set of training examples (training set) are generated where each training example is generated associated with one scheduling date. The above-mentioned features and labels are used to train the classifier model. A method of determining a label for each training example will now be described below according to various example embodiments of the present invention.

[0058] An optimal baseline resource schedule is first generated by carrying out an optimization across the whole scheduling horizon with the measured values of the stochastic parameters. This would generate the most optimal values (e.g., corresponding to optimal dispatch or allocation) of resources at each timestep as all uncertainties associated with the stochastic parameters are eliminated, and all information associated with the scheduling date are considered together during resource scheduling.

[0059] Next, real-time resource schedules are generated with rolling horizon optimization, where each resource schedule is generated with a possible choice of fixed prediction horizon lengths (e.g., if there are 12 possible choices of fixed prediction horizon lengths, then a total of 12 resource schedules are generated). In the rolling horizon optimization, an optimization is run at each timestep of the prediction horizon and only the first timestep's solutions are retained, the remaining solutions are discarded.

[0060] At each timestep of the real-time resource scheduling, the past measured values of the stochastic parameters are used to forecast their future values across the selected fixed prediction horizon. The forecasted values are used to run an optimization to minimize the cost objective (e.g., combined cost of resource cost and carbon emission cost).

[0061] Finally, the cost of each timestep across the scheduling horizon is determined for the optimal baseline resource schedule and for all the resource schedules generated for each fixed prediction horizon length. Thereafter, the RMSE between the timestep costs between the optimal baseline resource schedule and the resource schedules generated for each prediction horizon length are determined.

[0062] For the sake of clarity and better understanding, an illustrative example will now be described. Assume a scheduling horizon of 24 hours with an hourly timestep, and that the set of possible fixed prediction horizon lengths is 1 to 12. There will be 24 hourly values for the optimal baseline resource schedule and all 12 resource schedules generated for the set of 12 possible fixed prediction horizon lengths. For each of the 12 resource schedules generated, all 24 values of hourly cost are compared against the optimal baseline resource schedule's 24 values of hourly cost to calculate the RMSE value. As a result, a total of 12 RMSE values are obtained, each RMSE value obtained from the resource schedule generated with the corresponding fixed prediction horizon length.

[0063] In various example embodiments, the fixed prediction horizon length, whose generated resource schedule resulted in the lowest RMSE value, is deemed to be the most optimal prediction horizon length to use, and thus becoming the label of the training example. It is assumed that the determined optimal fixed prediction horizon length determined is optimal throughout the scheduling date's scheduling horizon.

[0064] In various example embodiments, the frequency of the use of the APH procedure in the ID scheduling horizon determines the number of classification models to be used. Using the illustrative example described hereinbefore, the APH procedure is implemented within the ID stage of the method 300. Across the 24 hours scheduling horizon, the APH procedure is activated after every four ID stage timestep t_{ID}, where the duration the ID stage timestep Δt_{ID} is one hour. Therefore, in this illustrative example, a total of five classification models may be utilized. Table 1 below lists the classification models and the timestep t_{ID} that each classification model is used at. Therefore, a RMSE value is calculated for each timestep t_{ID}. Accordingly, for this example, at $t_{ID} = 4$, 4 RMSE values are determined, at $t_{ID} = 8$, 8 RMSE values are determined, so on and so forth. This would lead to varying lengths of input to the classification model at timesteps $t_{ID} = 4, 8, 12, 16, 20$. Hence, in various example embodiments, different classification models are trained and used at timesteps $t_{ID} = 4, 8, 12, 16, 20$. In various example embodiments, the same classification model may be used for different scheduling dates, for example, Classification_model_1 is only used at $t_{ID} = 4$ on any scheduling dates, Classification_model_2 is only used at $t_{ID} = 8$ on any scheduling dates, and so on so forth.

Table 1 - Classification models for the APH procedure in explanation example

Classification model	Intraday Timestep Which its Used at (t_{ID})
Classification_model_1	4
Classification_model_2	8
Classification_model_3	12
Classification_model_4	16
Classification_model_5	20

[0065] There are no classification models for $t_{ID} = 0$ and $t_{ID} = 24$ as those are the beginning and end of the scheduling horizon, respectively. Each scheduling date is treated as independent, hence at the beginning of the scheduling horizon, there may be no available past timesteps' measured values for the stochastic parameters for the APH procedure to use to determine the optimal prediction horizon length. Instead, a prediction horizon length may be arbitrarily selected from the set of possible prediction horizon lengths. At $t_{ID} = 24$, it is the end of the scheduling horizon, and no further scheduling is performed. Hence, there is no need for the APH procedure to update the prediction horizon length to use.

[0066] For the training of each classification model, the classification model's features are obtained from all the past timesteps. For example, Classification_model_1 uses the past four timesteps information ($t_{ID} = 0, 1, 2, 3$), Classification_model_2 uses the past eight timesteps information ($t_{ID} = 0, 1, 2, 3 \ldots, 7$). This is also reflected in FIG. 4 which shows that the information from $t_{ID} = 0$ to $t_{ID} = \text{current K interval} - 1$ is used in the APH procedure to determine the optimal prediction horizon length. The classification model may be trained in this manner according to various example embodiments because, as described above, the number of inputs increase with the number of timesteps that has occurred. In this regard, various example embodiments note that a classification model that takes in multiple varying length input of features for prediction at various timestep present a technical challenge or problem, namely, it must be able to generalize across different 'timesteps considerations'. For example, if one model is trained that takes in the max number of inputs (e.g., which correspond to the input to Classification_model_5 that takes in 20 time steps information), at the earlier timestep, say for example $t_{ID} = 4$, the subsequent 16 timesteps must be masked (e.g., by setting them as infinity/-infinity). However, various example embodiments found that this will create very distant and isolated clusters of data points where each clusters are likely to be formed by data with similar number of non infinity/-infinity. It is therefore challenging for the classification algorithms to find the decision boundaries that can extend across the space between different

isolated clusters, and also achieve good decision boundaries within each cluster. To address this technical problem while avoiding sacrificing the classification accuracy performance, according to various example embodiments of the present invention, a respective classification model is used for each unique number of features fed as input, so that the decision boundaries only need to be formed within one 'isolated cluster' of data. Therefore, according to various example embodiments, different classification models are trained for optimizing the prediction horizon length over the scheduling horizon.

[0067] The assumption underlying the use of a supervised learning technique is that the patterns in the stochastic parameters learnt from the training set will be repeated in the future. This allows the APH procedure to identify the learnt patterns and determine the optimal prediction horizon length in real time. FIG. 5 depicts a flow diagram of the APH procedure according to various example embodiment of the present invention when implemented on an example real-time resource scheduling problem.

[0068] At the beginning of the resource scheduling, an arbitrary prediction horizon $PH_{initial}$ may be chosen from the specified range or predefined set of possible prediction horizon lengths (which is used to train the APH procedure's classifier as described hereinbefore). The rolling horizon optimization is employed. At each timestep, new forecasts of the stochastic parameters' future values are made across the current prediction horizon based on their past measured values. The forecasts are then used for optimization and only the solutions for the stochastic parameters of the first timestep is retained as the ID resource schedule's dispatch solution. At the start of every stipulated timestep interval K (i.e., t % K = 0, except the very first timestep interval), the prediction horizon length PH is updated with the optimal prediction horizon length determined by the APH procedure. The above APH procedure is repeated at the start of each timestep interval until the end of the scheduling horizon. The final resource schedule for the ID stage may thus be obtained by merging the determined values (the determined solutions) for the stochastic parameters retained for each first timestep of the rolling horizon optimization for all timesteps of the ID scheduling horizon.

[0069] The interstage re-optimization will now be described in further detail according to various example embodiments of the present invention.

[0070] The interstage re-optimization procedure is configured to determine if it is more optimal to follow the resource scheduling decisions made in the DA stage or to improve the DA resource schedule with the ID cost objective for optimizing solutions generated in the re-optimization performed in the ID stage. In various example embodiments, in the ID stage, a

rolling horizon optimization to minimize cost (e.g., combined resource cost and carbon emission costs) is performed across the ID scheduling horizon. At each timestep, past measured values of the stochastic parameters are used to generate new forecasts across the prediction horizon of the timestep. The new forecasted values are used to perform optimization across the prediction horizon (having a length determined and set by APH procedure) and only the first timestep's solutions for each optimization of the rolling horizon optimization across the ID scheduling horizon are retained. This strategy allows the ID stage to continually receive recent measured values of the stochastic parameters to generate more accurate forecasts and optimal solutions.

[0071] The interstage re-optimization procedure may operate in the follow manner according to various example embodiments of the present invention.

[0072] At the beginning of the resource scheduling, the values of weight parameters γ and δ (e.g., corresponding to the one or more optimization weight parameters described hereinbefore) are initialized with equal values where $\gamma = \delta = 0.5$, as no assumptions are made on the relative performance between the DA and ID stages.

[0073] After every stipulated timestep interval K, the RMSE of each stochastic parameters' past DA and ID forecasts values relative to the measured value are calculated, respectively (e.g., corresponding to the first and second error measures described hereinbefore). With the RMSE values, the relative errors between the past DA and ID forecasts are calculated. They are then used to determine the values of the weight parameters γ and δ in the ID stage's optimization problem's formulation. FIG. 6 depicts a flow diagram of the interstage re-optimization procedure according to various example embodiments of the present invention, which is configured to determine the values of the weight parameters γ and δ. The optimization function (e.g., corresponding to the second optimization function as described hereinbefore) at the ID stage may be expressed as follows:

$$\min_{P_{i,t}^{ID}} \gamma \cdot Deviation\ from\ DA\ Schedule + \delta \cdot Combined\ Cost\ of\ Re\text{-}optimised\ Schedule \quad \text{(Equation 1)}$$

where $P_{i,t}^{ID}$ denotes the decision variables representing the power dispatch of the various energy resources at current intraday timestep t_{ID}, and $i \in \{Energy\ Resource\}$.

[0074] With reference to FIG. 6, weight parameters γ_i and δ_i represent the set of weights first calculated for each stochastic parameter considered in the interstage re-optimization procedure. As illustrative examples, weight parameters γ and δ may be determined in the following manner, which is also illustrated in FIG. 6:

If $\left(RMSE_i^{DA} = RMSE_i^{ID}\right)$: $\quad where\ i \in \{Stochastic\ Parameters\}$

$$\gamma_i = 1,\ \delta_i = 0$$

(Equation 2)

Else: $\quad \gamma_i = 1 - \frac{RMSE_i^{DA}}{RMSE_i^{DA} + RMSE_i^{ID}}$

$$\delta_i = 1 - \gamma_i$$

(Equation 3)

$$\gamma = \frac{\sum_{i=1}^{N} \gamma_i}{N}$$

(Equation 4)

$$\delta = \frac{\sum_{i=1}^{N} \delta_i}{N}$$

(Equation 5)

where N is the total number of stochastic parameters considered.

[0075] Without wishing to be bound by theory but for better understanding, a rationale behind Equation (2) is that if the forecast errors in both the DA and ID stages are equal, the next consideration would then be the amount of information considered. Hence, the weight parameter γ_i is set to 1 to fully track the resource schedule generated in the DA stage as it has a longer prediction horizon. On the other hand, if the forecast errors are unequal, then, parameter weights γ_i and δ_i are determined based on their relative error, as expressed in Equation (2). Thereafter, a simple arithmetic average is done to obtain the final weights of γ and δ, as expressed in Equations (3) and (4), respectively.

[0076] For example, a larger value of parameter weight γ place more emphasis on minimizing the deviation from the DA stage's resource schedule, implying that it is more optimal to closely track the DA generated resource schedule. Conversely, a larger value of weight δ places greater emphasis on following the ID cost objective to obtain re-optimized solutions in the ID stage.

[0077] As illustrative examples, the two objective functions in the ID stage's optimization function may be expressed in Equations (6) and (7) below:

$$F_{Dev}^{ID} = \left| \sum_{t=1}^{T_{DA}=T_{ID}} P_{PV_t}^{DA} \nabla t_{DA} - \sum_{t=1}^{T_{ID}} P_{PV_t}^{ID} \nabla t_{ID} \right|$$

$$+\left|\sum_{t=1}^{T_{DA}=T_{ID}} P_{WTE_t}^{DA}\nabla t_{DA} - \sum_{t=1}^{T_{ID}} P_{WTE_t}^{ID}\nabla t_{ID}\right|$$
$$+\left|\sum_{t=1}^{T_{DA}=T_{ID}} P_{Grid_t}^{DA}\nabla t_{DA} - \sum_{t=1}^{T_{ID}} P_{Grid_t}^{ID}\nabla t_{ID}\right|$$
$$+\left|\sum_{t=1}^{T_{DA}=T_{ID}} P_{BESS,D_t}^{DA}\nabla t_{DA} - \sum_{t=1}^{T_{ID}} P_{BESS,D_t}^{ID}\nabla t_{ID}\right|$$
$$+\left|\sum_{t=1}^{T_{DA}=T_{ID}} P_{BESS,C_t}^{DA}\nabla t_{DA} - \sum_{t=1}^{T_{ID}} P_{BESS,C_t}^{ID}\nabla t_{ID}\right|$$

(Equation 6)

where F_{Dev}^{ID} denotes the deviation objective which quantifies the total deviation of the energy profile between the prediction horizon T_{ID} in the ID stage and the corresponding horizon in the DA stage's allocation plan. $P_{PV_t}^{DA}$, $P_{WTE_t}^{DA}$, $P_{Grid_t}^{DA}$ P_{BESS,D_t}^{DA}, P_{BESS,C_t}^{DA}, denote the allocation decisions of solar PV, waste-to-energy, power grid, BESS discharge and BESS charging made in the DA stage generated allocation plan. $P_{PV_t}^{ID}$, $P_{WTE_t}^{ID}$, $P_{Grid_t}^{ID}$, P_{BESS,D_t}^{ID}, P_{BESS,C_t}^{ID} denote the allocation decisions of solar PV, waste-to-energy, power grid, BESS discharge and BESS charging in the ID stage.

[0078] As the timesteps may be different between the DA stage and ID stage, the allocation decisions of each stage may be multiplied by their respective timestep interval, ∇t_{DA} and ∇t_{ID}, to obtain an energy profile for consistent comparison, for example:

$$F_{CC}^{ID} = (\alpha \cdot \omega \cdot F_{EC}^{ID} + \beta \cdot F_{RC}^{ID})$$

where,

$$F_{RC}^{ID} = \sum_{t=1}^{T_{ID}} RC_t^{ID} = \sum_{t=1}^{T_{ID}} \left(P_{PV_t}^{ID} \cdot RC_{PV} + P_{WTE_t}^{ID} \cdot RC_{WTE} + P_{Grid_t}^{ID} \cdot RC_{Grid_t} + P_{BESS,D_t}^{ID} \cdot RC_{BESS,D} + P_{BESS,C_t}^{ID} \cdot RC_{BESS,C}\right)$$

(Equation 7)

$$F_{EC}^{ID} = \sum_{t=1}^{T_{ID}} EC_t^{ID} = \sum_{t=1}^{T_{ID}} \left(P_{PV_t}^{ID} \cdot EC_{PV} + P_{WTE_t}^{ID} \cdot EC_{WTE} + P_{Grid_t}^{ID} \cdot EC_{Grid} + P_{BESS,D_t}^{ID} \cdot EC_{BESS,D} + P_{BESS,C_t}^{ID} \cdot EC_{BESS,C}\right)$$

(Equation 8)

where F_{CC}^{ID} denotes the cost objective which is the weighted sum of the resource cost F_{RC}^{ID} and carbon emission costs F_{EC}^{ID} over the prediction horizon T_{ID}. It has the same notations and meaning as the DA stage's optimization problem formulation. Accordingly, the cost objective at the ID stage is to run a re-optimization over T_{ID} with the new forecasts made with the real-time actual realised values of the uncertain parameters.

[0079] The weight parameters α and β are attached to the operating cost objective and the carbon emission cost objective, respectively. $\omega = 1$ is the conversion factor of carbon emission to the dollar amount. The values of the weight parameters determine the relative importance of each objective. They can be adjusted accordingly to prioritize either economic savings or carbon emission reduction.

[0080] Accordingly, the method 300 advantageously provides an interstage re-optimization mechanism implemented in the ID stage which is able to adjust the resource allocation plan generated at the DA stage. For example, the minimization objective of the ID stage optimization problem is the sum of the two objectives F_{Dev}^{ID} and F_{CC}^{ID}, which are weighted by weight parameters γ and δ respectively. These weight parameters are utilized to control the priority between following the generated allocation plan in the DA stage or adjusting the allocation plan through re-optimization with the updated forecasts based on the ID cost objective in the ID stage. This interstage re-optimization mechanism thus challenges the conventional understanding/assumption that the allocation plan generated in the DA stage is the most optimal due to a longer prediction horizon. Instead, it takes the forecast errors made in the recent past timesteps (e.g., hours) at the DA stage and compares them against the forecast errors made at the ID stage in the same period. A higher relative error in the DA stage means that the allocation plan was generated with less accurate forecasts in the DA stage. This implies that the DA stage's allocation plan may be sub-optimal, even though it is generated over a longer prediction horizon. Hence, according to various example embodiments, the adjusted allocation plan does not simply closely follow the DA stage's allocation plan and instead place more weight on the re-optimized allocation generated in the ID stage. This can be reflected with γ having a lower value than δ.

[0081] Accordingly, the resource scheduling method 300 according to various example embodiments of the present invention advantageously comprises adaptive prediction horizon (APH) and interstage re-optimization methods or modules. As described hereinbefore, the APH method may utilize a machine learning model, such as a classification model, trained to determine an optimal prediction horizon length in rolling horizon optimization at the ID stage. In this regard, at the start of every stipulated timestep interval (e.g., except the very first timestep interval) during resource scheduling at the ID stage, the optimal prediction horizon length may be determined using the classification model. The interstage re-optimization method is configured to determine if it is more optimal to follow the resource scheduling decisions made in the DA stage or to improve the resource schedule with the solutions generated by the ID cost

objective in the re-optimization performed in the ID stage. In various example embodiments, this is performed by determining the RMSE of each stochastic parameters' past DA and ID forecasts values relative to the measured values, respectively. With the RMSE values, the relative errors between the past DA and ID forecasts are calculated. They are then used to determine the weight parameters γ and δ for the ID stage's optimization problem's formulation. For example, a larger value of weight parameter γ places more emphasis on minimizing the deviation from the DA stage's resource schedule, implying that it is more optimal to closely track the DA generated resource schedule. Conversely, a larger value of parameter weight δ places greater emphasis on following the ID cost objective to obtain re-optimized solutions in the ID stage.

[0082] With the introduction of the above two modules or components, the resource scheduling method 300 is advantageously able to improve resource scheduling for enhancing the optimality of the resource schedule generated, thereby resulting in improved efficiencies and effectiveness in managing resources, such as enabling the reduction of deviation between the computed and retrospective allocation of energy and resources. For example, the method 300 is able to optimizes the real-time energy and resource allocation to meet the demand of any type of energy and resource consuming systems.

[0083] To demonstrate the improved efficiencies and effectiveness in managing resources by the resource scheduling method 300 according to various example embodiments of the present invention, the performance evaluation of the resource scheduling method 300 simulated with an example case study using the Model Factory at SIMTech. As an example, the resource scheduling method 300 was simulated on 7 scheduling dates ranging from 1st to 7th September. This work's proposed real-time multistage scheduling algorithm performance is compared against other energy scheduling algorithms in the literature to analyze its relative performance. The two algorithms adopted for comparison in this work include a three-stage energy scheduling algorithm from Rana *et al.*, "Real-Time Scheduling of Community Microgrid," *J. Clean. Prod.*, vol. 286, 125419, 2021 (hereinafter referred to as the Rana reference), which deals with energy management in a community microgrid, and a two-stage energy scheduling algorithm from Jeong *et al.*, "Implementation of Optimal Two-Stage Scheduling of Energy Storage System Based on Big-Data-Driven Forecasting – An Actual Case Study in a Campus Microgrid", *Energies*, vol. 12, no. 6, Art. no. 6, Jan. 2019 (hereinafter referred to as the Jeong reference), which was implemented on a campus microgrid. Modifications were made to the adopted reference algorithms to contextualize them in this work. Caution was exercised to

ensure that the broad concept of their algorithms is retained. The algorithms were benchmarked against an optimal baseline resource allocation plan, which is generated retrospectively with all uncertain parameters' values known.

[0084] Table 2 shown in FIG. 7A summarizes the parameters and constraint values that were used in the generation of the allocation plan used across all the algorithms in this simulation. Table 3 shown in FIG. 7B summarizes information on the three resource scheduling algorithms. In the DA stage of all algorithms, prediction of the uncertain parameters across the whole scheduling day is performed. The forecasted information is used to generate an hourly allocation plan over 24 hours. For the three-stage algorithms, predictions are carried out over the prediction horizon with the real-time realized values of uncertain parameters at the second stage. In this work, fixed prediction horizons are between 1 to 12 hours long, with an hourly timestep assigned to the ID stage. Finally, in the RT stage of all algorithms, no forecast is made, hence there is no prediction horizon. Instead, the real-time realized values of uncertain parameters are received at each 15 minutes timestep and used to adjust the dispatch allocation decisions.

[0085] Table 4 shown in FIG. 7C shows the average deviation (per timestep) across all power resource allocations from the optimal baseline for the different scheduling algorithms. The table categorizes the deviations based on the different fixed prediction horizon lengths used in the intraday stage. The deviation is calculated by comparing the power source allocation decision at each (real-time) timestep across all seven days of the scheduling dates. Mean Absolute Error (MAE) is used to quantify the power allocation deviation from the optimal baseline.

[0086] As observed in Table 7 in FIG. 7C, the MAE of the present real-time multistage scheduling algorithm's power allocation from the optimal baseline is consistently lower than the other algorithms across all fixed prediction horizon lengths used in the ID stage. It is expected that a 3-stage algorithm should perform better than a 2-stage algorithm. The additional ID stage in 3-stage algorithms obtains updated real-time information and forecasts which are used to adjust allocation decisions made in the earlier generated allocation plan in the DA stage. This reduces the uncertainty and generates a more optimal allocation plan. The present 3-stage algorithm aligns with such an understanding. However, the Rana reference's 3-stage algorithm performs worse than the 2-stage algorithm of the Jeong reference. Another observation made is that the use of different fixed prediction horizons in the ID stage has an impact on the optimality

of the generated allocation plan. However, there is no consistent trend observed on the magnitude of the MAE across the fixed prediction horizon used.

[0087] In the following comparisons of the different cumulative costs across the scheduling dates, the cost index metric is used. This is to facilitate clear comparison. With the cost index, the optimal baseline allocation plan's cost index will have a base value of 1 and the respective algorithms' cost index is interpreted as the amount incurred for every unit cost incurred in the optimal baseline allocation plan.

[0088] Based on the combined cost index in Table 5 shown in FIG. 7D, the present real-time multistage scheduling algorithm according to various example embodiments of the present invention consistently has the least deviation from the optimal baseline as compared to the other two comparison algorithms. Furthermore, in terms of combined costs, the present real-time multistage scheduling algorithm has up to 6.45% and 4.98% improvement from the Rana reference's algorithm and the Jeong reference's algorithm respectively.

[0089] The combined cost of the generated allocation plans can be further split into resource cost and carbon emission cost. To get a better picture of whether the observed improvement of the combined cost by the present real-time multistage scheduling algorithm is attributed to the resource cost or carbon emission cost, the respective costs are analyzed separately.

[0090] Looking at the resource cost index in Table 6 shown in FIG. 7E, the present real-time multistage scheduling algorithm has the least deviation from the optimal baseline. the proposed novel algorithm has up to 6.76% improvement from the Rana reference's algorithm and up to 5.86% improvement from the Jeong reference's algorithm in terms of resource costs.

[0091] From Table 7 shown in FIG. 7F, the present real-time multistage scheduling algorithm has the lowest deviation from the optimal baseline allocation plan in terms of carbon emission cost. The present real-time multistage scheduling algorithm has an improvement of up to 7.28% and 4.58% from the Rana reference's algorithm and the Jeong reference's algorithm respectively.

[0092] It is observed that while the Rana reference's algorithm generates allocation plans with lower resource cost than the Jeong reference's, its carbon emission cost is greater. This suggests that the Rana reference's algorithm could be allocating more of the energy resource which has lower resource cost but higher carbon emission cost. Ultimately, in the Rana reference's algorithm, the carbon emission cost outweighs the savings derived from the lower resource cost and drives up the combined cost, leading it to have the highest combined cost index over the other algorithms. This suggests that the Rana reference's algorithm is better than

the Jeong reference's algorithm at fulfilling the objective of minimizing resource cost but does not perform as well with carbon emission cost minimization. Based on the above results, it can be concluded that the real-time multistage scheduling algorithm according to various example embodiments of the present invention is superior to the other two algorithms in fulfilling the minimization objectives of both the resource cost and carbon emission cost.

[0093] Another observation made is that the use of different fixed prediction horizons in the lD stage impacts the optimality of the generated allocation plan. Hence, across a particular scheduling period, there is no one best prediction horizon length to use to generate an optimal allocation plan. This demonstrates the need to determine an optimal prediction horizon length adaptively or dynamically across the scheduling horizon as achieved by the APH procedure according to various example embodiments of the present invention.

[0094] While embodiments of the invention have been particularly shown and described with reference to specific embodiments, it should be understood by those skilled in the art that various changes in form and detail may be made therein without departing from the scope of the invention as defined by the appended claims. The scope of the invention is thus indicated by the appended claims and all changes which come within the meaning and range of equivalency of the claims are therefore intended to be embraced.

CLAIMS

1. A method of multistage resource scheduling, the method comprising:

determining, at a first stage, a first stage resource schedule across a first stage scheduling horizon, comprising a series of timesteps, for a plurality of resource parameters based on a first optimization function; and

determining, at a second stage, a second stage resource schedule across a second stage scheduling horizon, comprising a series of timesteps, for the plurality of resource parameters based on a second optimization function, comprising, for each timestep, in turn, of the series of timesteps of the second stage scheduling horizon:

determining a resource schedule across a prediction horizon with respect to the timestep, comprising a subseries of timesteps, starting with the timestep, of the series of timesteps of the second stage scheduling horizon, for the plurality of resource parameters based on the second optimization function; and

storing values of the resource schedule determined for the plurality of resource parameters for a beginning timestep of the subseries of timesteps as determined values for the plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon,

wherein for each of one or more timesteps or each of one or more intervals of timesteps of the series of timesteps of the second stage scheduling horizon, respectively, the method further comprises determining and setting a length of the prediction horizon for said determining the resource schedule across the prediction horizon.

2. The method according to claim 1, wherein

for said each timestep, in turn, of the series of timesteps of the second stage scheduling horizon:

said determining the resource schedule across the prediction horizon with respect to the timestep comprises generating forecasted values for the subseries of timesteps of the prediction horizon for the plurality of resource parameters, and

the method further comprises storing forecasted values for the plurality of resource parameters for the beginning timestep of the subseries of timesteps as forecasted values for the plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon, and

for said each of the one or more timesteps or said each of the one or more intervals of timesteps of the second stage scheduling horizon, respectively, the length of the prediction horizon for said determining the resource schedule across the prediction horizon is determined based on an error measure between past forecasted values stored and measured values for the plurality of resource parameters for one or more past timesteps of the second stage scheduling horizon.

3. The method according to claim 2, wherein the length of the prediction horizon for said determining the resource schedule across the prediction horizon is determined using a prediction horizon length machine learning model based on the error measure between the past forecasted values stored and the measured values for the plurality of resource parameters for the one or more past timesteps of the second stage scheduling horizon.

4. The method according to claim 1, wherein

for said each timestep, in turn, of the series of timesteps of the second stage scheduling horizon, the second optimization function comprises a schedule deviation optimization function and a resource cost optimization function,

the second optimization function is configured to minimize, over the prediction horizon with respect to the timestep, a deviation between the resource schedule determined across the prediction horizon with respect to the timestep and a corresponding portion of the first stage resource schedule, and

the resource cost optimization function is configured to minimize, over the prediction horizon with respect to the timestep, a cost objective associated with the plurality of resource parameters.

5. The method according to claim 4, wherein

said determining, at the first stage, the first stage resource schedule across the first stage scheduling horizon comprises generating and storing forecasted values for the series of timesteps of the first stage scheduling horizon for the plurality of resource parameters,

for said each timestep, in turn, of the series of timesteps of the second stage scheduling horizon:

said determining the resource schedule across the prediction horizon with respect to the timestep comprises generating forecasted values for the subseries of timesteps of the prediction horizon for the plurality of resource parameters, and

the method further comprises storing forecasted values for the plurality of resource parameters for the beginning timestep of the subseries of timesteps as forecasted values for the plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon,

for said each of one or more timesteps or said each of one or more intervals of timesteps of the second stage scheduling horizon, the method further comprises:

determining and setting one or more values of one or more optimization weight parameters, respectively, based on past forecasted values stored for the plurality of resource parameters for the first stage and past forecasted values stored for the plurality of resource parameters for the second stage, wherein the second optimization function is weighted by the one or more optimization weight parameters.

6. The method according to claim 5, wherein

the one or more values of the one or more optimization weight parameters are determined based on a first error measure and a second error measure,

the first error measure is between the past forecasted values stored and measured values for the plurality of resource parameters for one or more past timesteps of the first stage scheduling horizon, and

the second error measure is between the past forecasted values stored and measured values for the plurality of resource parameters for the one or more past timesteps of the second stage scheduling horizon.

7. The method according to claim 6, wherein the second optimization function is weighted by the one or more optimization weight parameters such that a larger first error measure relative to the second error measure results in a larger weighting of the resource cost optimization function relative to the schedule deviation optimization function, and a smaller first error measure relative to the second error measure results in a smaller weighting of the resource cost optimization function relative to the schedule deviation optimization function.

8. The method according to claim 1, further comprising:

determining, at a third stage, a third stage resource schedule across a third stage scheduling horizon, comprising a series of timesteps, for the plurality of resource parameters based on a third optimization function, the third optimization function being configured to minimize a deviation between the third stage resource schedule and the second stage resource schedule subject to one or more constraints.

9. The method according to claim 8, wherein the first stage, the second stage and the third stage correspond to a day-ahead stage, an intraday stage and a real-time stage, respectively.

10. The method according to claim 5, wherein said determining and setting the length of the prediction horizon and said determining and setting the one or more values of the one or more optimization weight parameters are performed at a beginning of each interval of a plurality of consecutive intervals of timesteps of the series of timesteps of the second stage scheduling horizon.

11. A system for multistage resource scheduling, the system comprising:

at least one memory; and

at least one processor communicatively coupled to the at least one memory and configured to:

determine, at a first stage, a first stage resource schedule across a first stage scheduling horizon, comprising a series of timesteps, for a plurality of resource parameters based on a first optimization function; and

determine, at a second stage, a second stage resource schedule across a second stage scheduling horizon, comprising a series of timesteps, for the plurality of resource parameters based on a second optimization function, comprising, for each timestep, in turn, of the series of timesteps of the second stage scheduling horizon:

determining a resource schedule across a prediction horizon with respect to the timestep, comprising a subseries of timesteps, starting with the timestep, of the series of timesteps of the second stage scheduling horizon, for the plurality of resource parameters based on the second optimization function; and

storing values of the resource schedule determined for the plurality of resource parameters for a beginning timestep of the subseries of timesteps as determined values for the

plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon,

wherein for each of one or more timesteps or each of one or more intervals of timesteps of the series of timesteps of the second stage scheduling horizon, respectively, the at least one processor is further configured to determine and set a length of the prediction horizon for said determining the resource schedule across the prediction horizon.

12. The system according to claim 11, wherein

for said each timestep, in turn, of the series of timesteps of the second stage scheduling horizon:

said determining the resource schedule across the prediction horizon with respect to the timestep comprises generating forecasted values for the subseries of timesteps of the prediction horizon for the plurality of resource parameters, and

the at least one processor is further configured to store forecasted values for the plurality of resource parameters for the beginning timestep of the subseries of timesteps as forecasted values for the plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon, and

for said each of the one or more timesteps or said each of the one or more intervals of timesteps of the second stage scheduling horizon, respectively, the length of the prediction horizon for said determining the resource schedule across the prediction horizon is determined based on an error measure between past forecasted values stored and measured values for the plurality of resource parameters for one or more past timesteps of the second stage scheduling horizon.

13. The system according to claim 12, wherein the length of the prediction horizon for said determining the resource schedule across the prediction horizon is determined using a prediction horizon length machine learning model based on the error measure between the past forecasted values stored and the measured values for the plurality of resource parameters for the one or more past timesteps of the second stage scheduling horizon.

14. The system according to claim 11, wherein

for said each timestep, in turn, of the series of timesteps of the second stage scheduling horizon, the second optimization function comprises a schedule deviation optimization function and a resource cost optimization function,

the second optimization function is configured to minimize, over the prediction horizon with respect to the timestep, a deviation between the resource schedule determined across the prediction horizon with respect to the timestep and a corresponding portion of the first stage resource schedule, and

the resource cost optimization function is configured to minimize, over the prediction horizon with respect to the timestep, a cost objective associated with the plurality of resource parameters.

15. The system according to claim 14, wherein

said determine, at the first stage, the first stage resource schedule across the first stage scheduling horizon comprises generating and storing forecasted values for the series of timesteps of the first stage scheduling horizon for the plurality of resource parameters,

for said each timestep, in turn, of the series of timesteps of the second stage scheduling horizon:

said determining the resource schedule across the prediction horizon with respect to the timestep comprises generating forecasted values for the subseries of timesteps of the prediction horizon for the plurality of resource parameters, and

the at least one processor is further configured to store forecasted values for the plurality of resource parameters for the beginning timestep of the subseries of timesteps as forecasted values for the plurality of resource parameters for the timestep of the series of timesteps of the second stage scheduling horizon,

for said each of one or more timesteps or said each of one or more intervals of timesteps of the second stage scheduling horizon, the at least one processor is further configured to:

determine and set one or more values of one or more optimization weight parameters, respectively, based on past forecasted values stored for the plurality of resource parameters for the first stage and past forecasted values stored for the plurality of resource parameters for the second stage, wherein the second optimization function is weighted by the one or more optimization weight parameters.

16. The system according to claim 15, wherein

the one or more values of the one or more optimization weight parameters are determined based on a first error measure and a second error measure,

the first error measure is between the past forecasted values stored and measured values for the plurality of resource parameters for one or more past timesteps of the first stage scheduling horizon, and

the second error measure is between the past forecasted values stored and measured values for the plurality of resource parameters for the one or more past timesteps of the second stage scheduling horizon.

17. The system according to claim 16, wherein the second optimization function is weighted by the one or more optimization weight parameters such that a larger first error measure relative to the second error measure results in a larger weighting of the resource cost optimization function relative to the schedule deviation optimization function, and a smaller first error measure relative to the second error measure results in a smaller weighting of the resource cost optimization function relative to the schedule deviation optimization function.

18. The system according to claim 11, wherein the at least one processor is further configured to:

determine, at a third stage, a third stage resource schedule across a third stage scheduling horizon, comprising a series of timesteps, for the plurality of resource parameters based on a third optimization function, the third optimization function being configured to minimize a deviation between the third stage resource schedule and the second stage resource schedule subject to one or more constraints.

19. The system according to claim 18, wherein the first stage, the second stage and the third stage correspond to a day-ahead stage, an intraday stage and a real-time stage, respectively.

20. The system according to claim 15, wherein said determine and set the length of the prediction horizon and said determine and set the one or more values of the one or more optimization weight parameters are performed at a beginning of each interval of a plurality of consecutive intervals of timesteps of the series of timesteps of the second stage scheduling horizon.